(12) UK Patent Application (19) GB (11) 2 297 636 (13) A

(43) Date of A Publication 07.08.1996

(21) Application No 9502377.6

(22) Date of Filing 02.02.1995

(71) Applicant(s)
Spring Consultants Limited

(Incorporated in the United Kingdom)

Unit 5, Ashbrook Mews, Westbrook Street,
BLEWBURY, Oxon, OX11 9QA, United Kingdom

(72) Inventor(s)
Norman Hamilton Burkles
Andrew Paul George Randall

(74) Agent and/or Address for Service
Atkinson & Co
Sixth Floor,High Holborn House, 52-54 High Holborn,
LONDON, WC1V 6SE, United Kingdom

(51) INT CL^6
G06F 3/06

(52) UK CL (Edition O)
G4A AFS AMX

(56) Documents Cited
EP 0078683 A2
Dialog record 01425541 of UNIX Review, vol. 9, No.4,
April 1991, page 98

(58) Field of Search
UK CL (Edition N) G4A AFS AMX
INT CL^6 G06F 3/06
ONLINE : WPI, INSPEC, COMPUTER DATABASE

(54) Storing data on emulated, logical, removable, disc drives

(57) Data is stored on a large storage volume implemented as a redundant array of five inexpensive discs (21-25). This volume is controlled so as to emulate the presence of a plurality of logical drives. Workstations (15,16) accessing the drives perceive them as removable SCSI drives. Consequently, when a remote workstation closes access to a previously accessed logical drive, a disc dismount command is generated, as required by a removable disc drive, thereby enabling other workstations to obtain access to that drive.
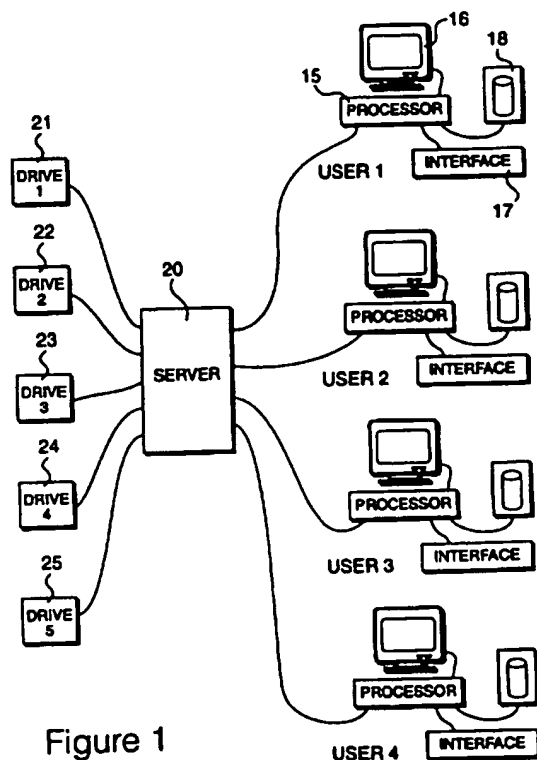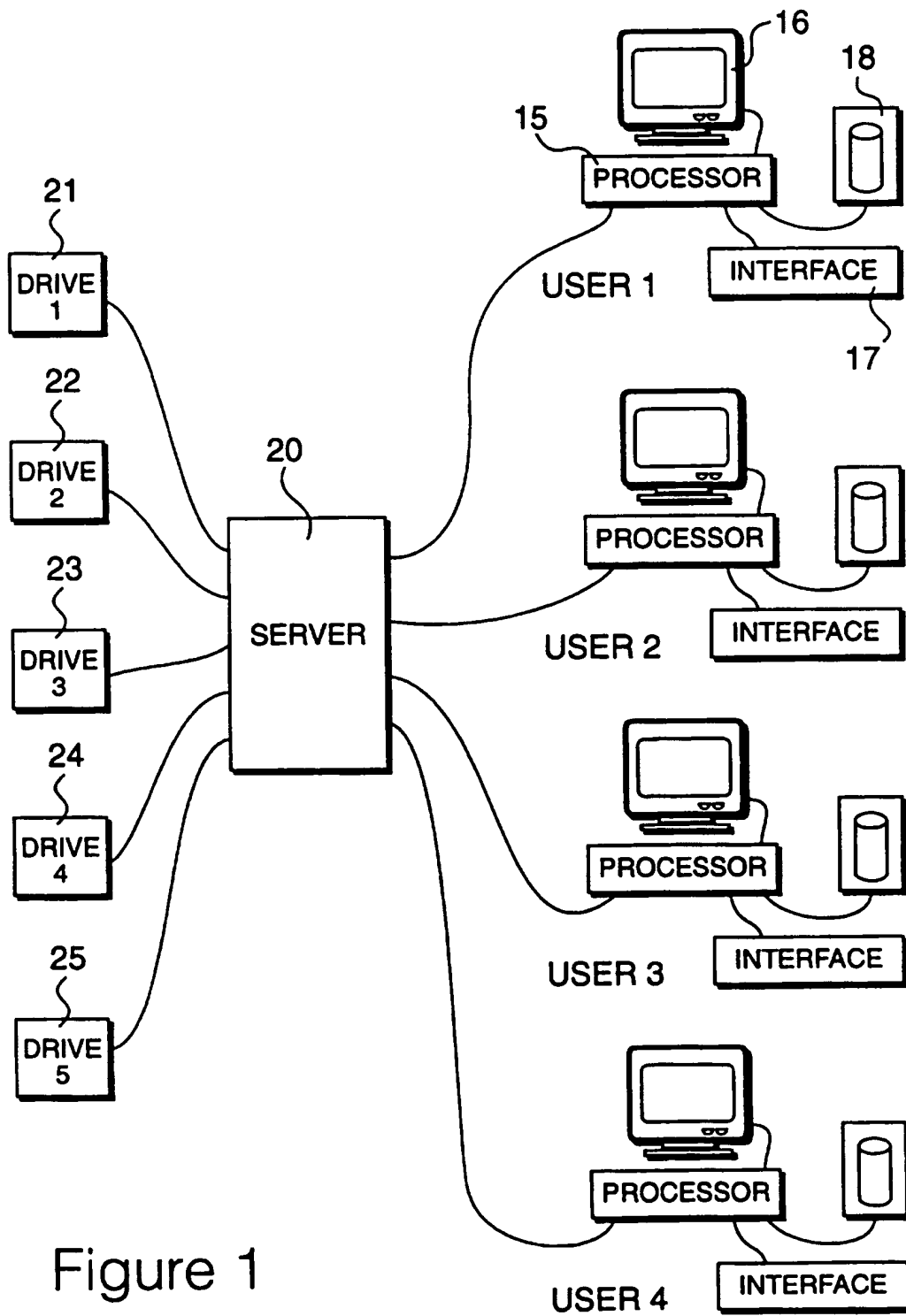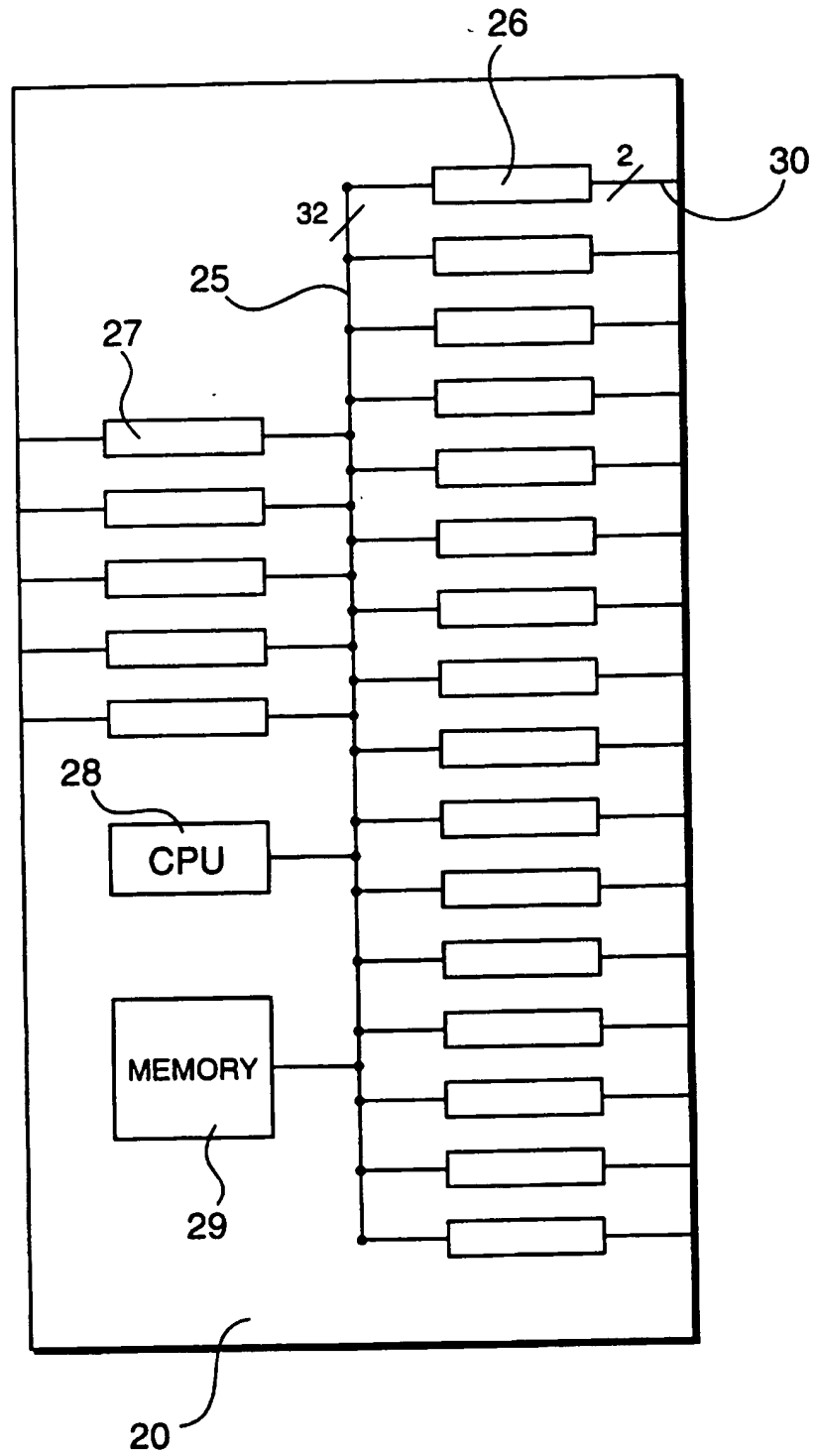

Figure 1

GB 2 297 636 A

Figure 1

Figure 2

USER IDENTIFIES LOGICAL DRIVE —31

LOCAL OPERATING SYSTEM REQUESTS ACCESS —32

SERVER GRANTS ACCESS AND RETURNS SIZE AND TYPE —33

DATA TRANSFER —34

USER TERMINATES ACCESS —35

WORKSTATION ISSUES DISMOUNT COMMAND —36

NETWORK ACTS UPON DISMOUNT COMMAND —37

WAIT FOR NEXT USER COMMAND —38

Figure 3

LOCAL SECTOR
SELECTION

61

62

63

USER
INTERFACE

LOCAL
OPERATING
SYSTEM

SCSI
INTERFACE

FILE SELECTION

SERVER
INTERFACE

SERVER
OPERATING
SYSTEM

PHYSICAL
DISC

TRANSFER
OF DATA

65

SERVER BLOCK
SELECTION

66

67

68

SECTOR
SELECT

TABLE

69

ADDRESSED BLOCKS

Figure 4

START

71 — WAIT FOR WORKSTATION COMMAND

EMULATE EMPTY DRIVE
—73

72 — INSTRUCTED TO MOUNT ? —NO→

YES

REPLY "NOT READY"
—75

74 — DRIVE FREE ? —NO→

YES

76 — ALLOCATE PARTITION

77 — REPLY "MOUNTED"

78 — WAIT FOR COMMANDS

EMULATE DISC
—80

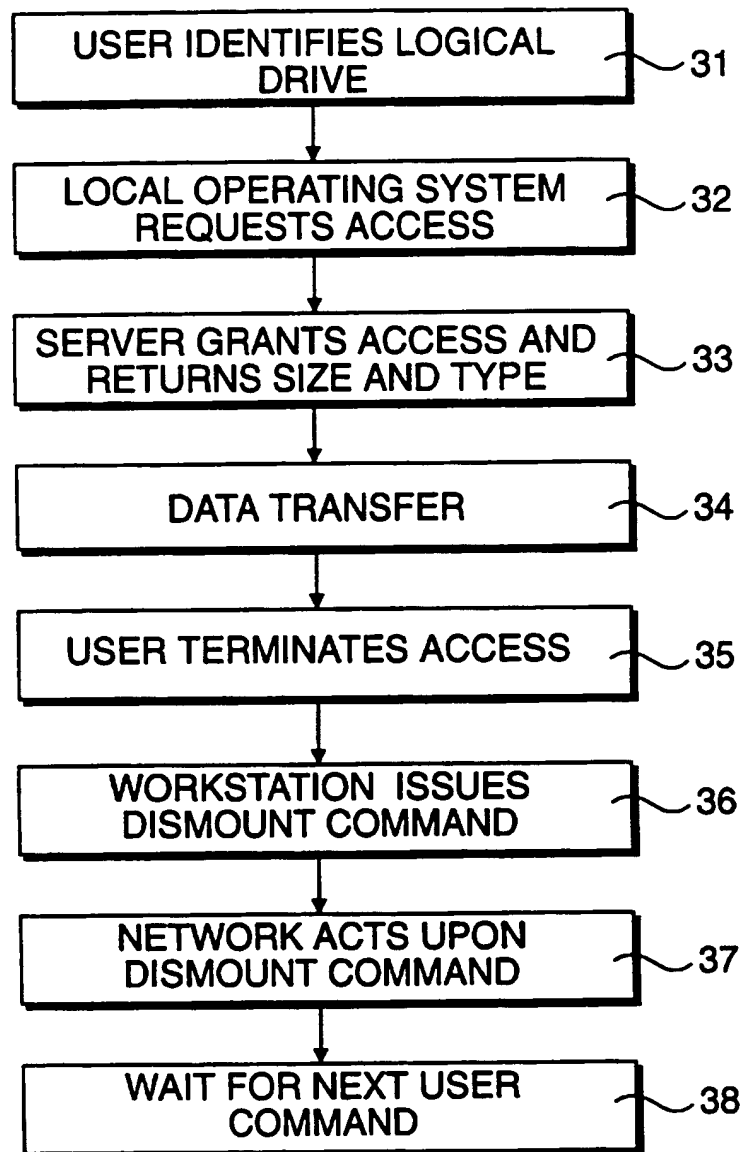79 — DISMOUNT ? —NO→

YES

81 — DE-ALLOCATE PARTITION

82 — REPLY "DISMOUNTED"
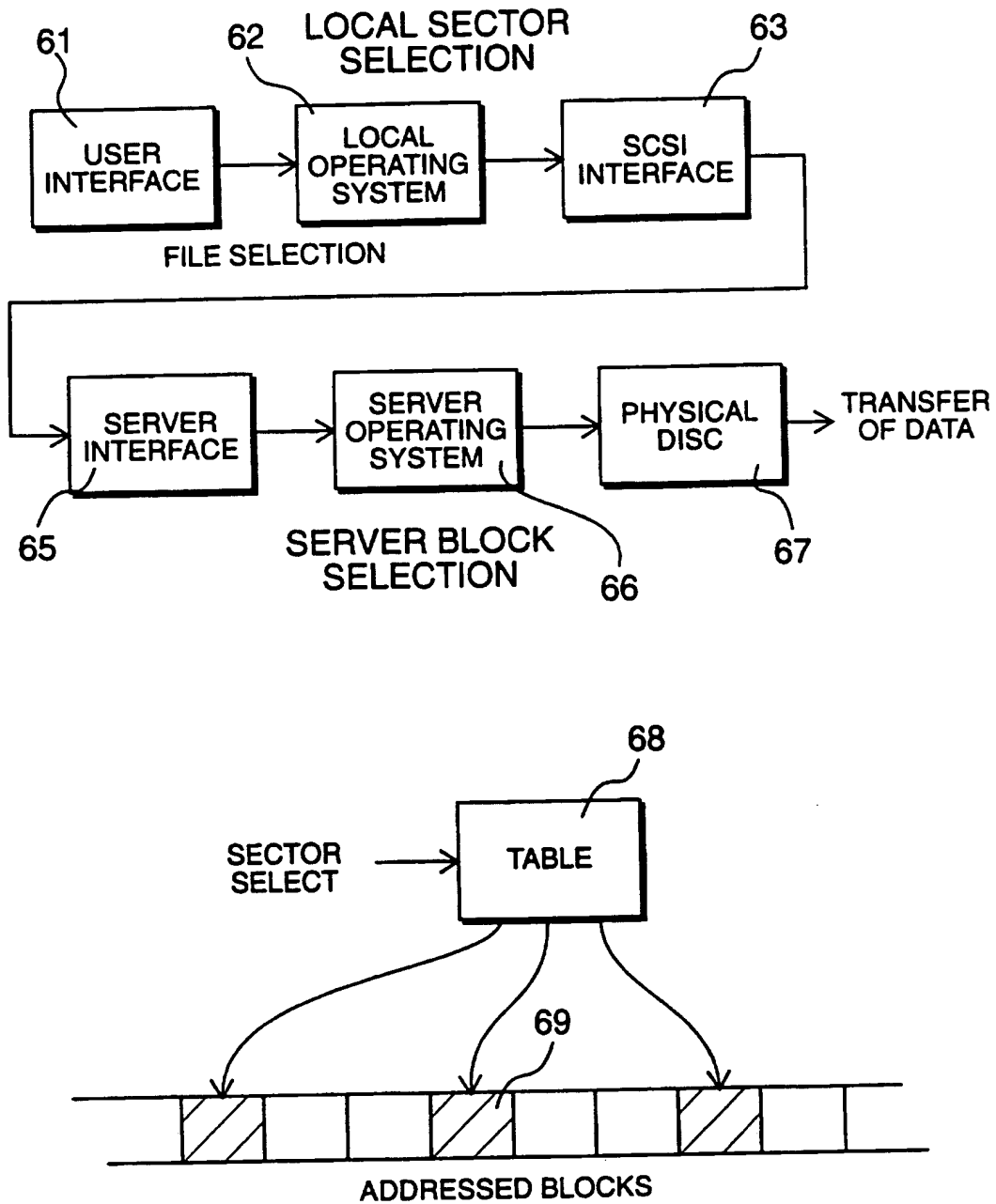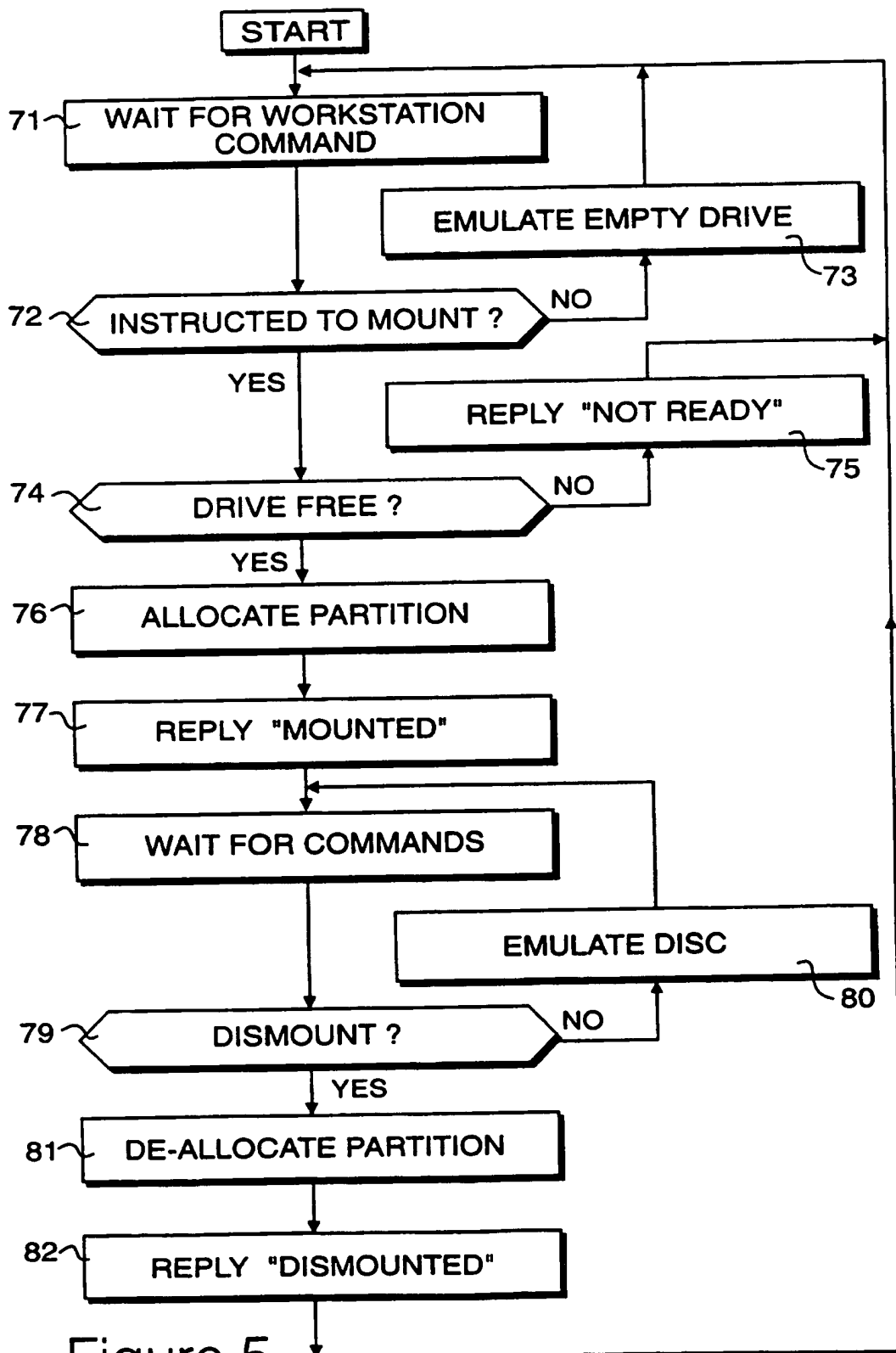
Figure 5

STORING DATA

The present invention relates to storing data. In particular, the present invention relates to large storage volumes controlled so as to emulate the presence of a plurality of logical drives.

Systems are known in which a large storage volume emulates the presence of a plurality of smaller volumes, which in turn may assist a user by facilitating logical arrangement of data, such that data of a first type may be kept separate from data of a second type. As far as an operating system is concerned, it has access to a plurality of drives as an alternative to having access to only one drive. Most operating systems are capable of controlling a plurality of logical drives in this way; within limits.

In more sophisticated environments, it is possible for a plurality of users to be given access to a shared volume divided into a plurality of logical drives. The division of the volume into a plurality of logical drives facilitates the interchange of information between users. Thus, a first user may log onto a logical drive, manipulate data contained within that drive and then log off, so as to allow another user to be given access to the logical drive. Such a procedure is particularly attractive when large data files are being handled, such as data files representing full colour graphic images, where the transfer of data, even over relatively fast networks, may take a considerable amount of time.

In addition, a large shared volume may be constructed first to provide relatively fast access times, along with levels of redundancy, such that a

single destructive event would not result in the whole data being lost, with recovery procedures being included as part of the overall structure.

Increasingly, computer workstations are being provided with localised processing capabilities having recognised and well supported operating systems. Examples are Apple Macintosh computers, IBM personal computers and Unix workstations etc. All of these systems have recognised protocols for the transfer of data. Thus, given the abundance of well supported operating systems, it is preferable to take full advantage of these operating systems so as to minimise the degree of bespoke software which needs to be generated and subsequently supported. System designs are restricted if full adherence to existing standards must be maintained, however, in some environments, an established system of operation may already be functional and the extent to which this system may be modified by the addition of new software etc., may be severely restricted. In some situations, the installation of a new suite of networking software may invalidate software agreements relating to primary localised processing.

In an environment in which a large storage volume emulates a plurality of discs, contention problems occur and the control processor must ensure that strict housekeeping routines are maintained, such that, for example, a previously accessed logical drive is properly deactivated when a particular user has finished with it, so that said drive may be accessed by other users and the overall integrity of this system is maintained. However, the degree to which network software requires to be embedded within workstation software should be minimised and it is undesirable for the network to place additional constraints on the workstations so as to assist the network's processing devices with their housekeeping tasks.

According to a first aspect of the present invention, there is provided a method of storing data, wherein a large storage volume emulates a plurality of logical drives; said logical drives emulate removable disc drives; and the closing of access to a previously accessed logical drive generates a disc

5   dismount command.

Thus, an advantage of the present invention is that the logical drives emulated by the large storage volume are presented to users in the form of removable disc drives, although in preferred practical realisations, they would actually be embodied within an environment of large fixed drives, so as to

10   optimise data capacity and disc access speed. However, operating systems for the individual workstations are fully conversant with the requirements of removable disc drives and, as required by the present invention, they will issue commands to said drives, informing the drive that access is no longer required.

15   In this way, it is possible to ensure that all necessary housekeeping procedures are effected when control over a logical disc drive is relinquished, either as part of normal operations or due to a software or hardware fault. Thus, for example, it is possible to ensure that directory information, cached in memory, is written back to disc, thereby updating the disc's directory,

20   before releasing access to the logical drive.   Thus, by emulating removable drives of this type, workstation software will automatically provide the necessary levels of housekeeping in order to ensure that access to a logical drive is released when no longer required by a particular operator.

The local workstation will interface with a logical drive over standard

25   interfaces, provided for accessing removable disc drives. The workstation software will generate a disc dismount command and as far as the said

software is concerned, a dismount of the removable disc will be effected, thereby releasing the tie between the local workstation and that particular logical disc drive. However, within the network, this command will be interpreted to the effect that the processor no longer requires access to the

5   logical drive, thereby allowing housekeeping procedures to be performed by the network processor.

Preferably, the logical drives emulate removable SCSI drives which may be capable of storing between 200 MBytes and 900 MBytes of data. According to a second aspect of the present invention there is provided

10   apparatus, including a large storage volume; a control device arranged to control data transfer with said storage volume and to provide user terminal access to said storage volume by emulating the presence of a plurality of removable disc drives wherein user terminals generate a disc dismount command when closing access to a previously accessed logical drive; and the

15   control device responds to said disc dismount command by terminating connection to said previously connected logical drive.

In a preferred embodiment, the control device is arranged to read directory information from an access logical drive and said directory information stored on the disc is updated in response to a disc dismount

20   command.

The invention will now be described by way of example only, with reference to the accompanying figures, in which:

Figure 1 shows a system in which a plurality of workstations have access to a shared storage volume, including a file server;

Figure 2 details the file server shown in Figure 1;

Figure 3 details operations performed by the system shown in Figure 1; and

Figure 4 represents the logical operations effected by the system shown in Figure 1, including removable disc emulation;

Figure 5 details the removable disc emulation procedures performed by the file server shown in Figure 1.

A system is shown in Figure 1 in which a plurality of users have access to a shared storage volume. At each user workstation, the user is provided with a processor 15, a visual display unit 16, a keyboard, mouse or similar interface device 17 and a local disc drive 18.

Each processor 15 includes conventional software so as to implement an operating system, allowing data transfer between the processor 15 and the disc drive 18. In addition, the operating system also facilitates data transfer between the processors 15 and a shared file server 20. In this preferred embodiment, the file server 20 is connected to five physical hard disc drives 21, 22, 23, 24 and 25, which in combination provide a total of thirty-six GBytes of storage with an access speed of typically 10 MBytes per second.

Disc drives 21 to 25 are configured as a redundant array, in which actual data is stored on four of the drives, with parity data stored on the fifth. In this way, any one of the physical drives 21 to 25 may be removed from the system, possibly due to operational failure (head crash etc.) whereafter said data may be re-constituted from the data available from the other four.

Thus, data integrity and reliability are assured without the need for implementing regular back-up procedures. The use of a plurality of disc drives in this way is known in the art as a redundant array of inexpensive discs. In the preferred embodiment this is implemented in accordance with

5   the RAID 5 recommendation.

Data is written to the drives in the form of identifiable blocks or regions of a predetermined length. The size of these blocks is determined from a trade-off between disc space optimisation and disc fragmentation. The system is primarily designed for storing large full colour graphics files and

10   blocks have a size of, typically, between two MBytes and thirty-two MBytes, although block size may be configurable so as to suit particular applications. In operation, users issue commands under software control which result in logical drives being made available by the server 20. Communication between users and the server 20 is implemented using established protocols.

15   In the preferred embodiment, the standard small computer systems interface (SCSI) is implemented and suitable interface cards are mounted in association with processor 15 and server 20. Thus, once a logical drive has been established by the server 20, this drive may be accessed by the user who perceives the drive as a conventional SCSI drive, accessed via conventional

20   protocols within the local operating system.

The server 20 is arranged to provide access to a total of sixteen user workstations and a further sixteen workstations may be given access by connecting a similar server in tandem with the first. The server is detailed in Figure 2 and, internally, a thirty-two bit parallel bus 25 provides

25   communication between the user interface circuits 26 and disc drive interfaces 27. The server is controlled in response to commands issued by the central

processing unit 28 which in turn receives programmed instructions from an internal memory device 29.

As previously stated, the server 20 is connected to each processor of a user workstation via a SCSI interface. The range of such interfaces is

5 limited and in alternative embodiments it may be necessary to provide alternative connections, possibly via coaxial cables, so as to increase the distance between the server and the workstations. It is therefore envisaged that systems will be designed specifically for particular applications, so as to optimise connections between workstations and the server. Thus, in some

10 environments, a large number of workstations may be provided relatively close to the server 20, in which case conventional SCSI interfaces may be employed whereas, in alternative arrangements, workstations may be distributed quite widely throughout a building, requiring more robust connections between the processors and the server 20. It is envisaged that

15 connections of this type should allow the workstations to be displaced from the server by distances in excess of 100 metres, having characteristics similar to high speed ethernet links.

Typical operation of the system shown in Figure 1 is detailed in Figure 3. As far as the operating system executable by each user workstation is

20 concerned, the workstation effectively has access to a large number of removable disc drives, although these are actually emulated by the server 20. In some situations, standard operating system software interfaces may be implemented within the user workstations so as to allow users to gain access to these logical drives. However, as the number of logical drives increases,

25 it may be necessary to improve the environment provided for users, so that they are aware of the presence of the disc drives and are provided with an interface which facilitates access to them. However, these user interfaces

would be overlaid over the operating system so that computer generated commands would result in instructions being generated at the operating system level.

5   Referring to Figure 3, a user identifies a logical disc drive to which access is required and identifies this logical disc drive at step 31. In response to the local request made at step 31, the local operating system implements measures to effect a request to access the logical disc drive, using conventional protocols. In particular, the processor 15 issues commands over the SCSI interface connected to the server 20.

10   In response to the request made at step 32, the server 20 will determine whether the logical disc drive is available and if the drive is available, it will grant access to the requesting workstation. As part of the SCSI protocol, the server will return data back to the requesting workstation, identifying the size of the logical drive and the drive type. Data relating to the drive type is very

15   relevant to the present invention. In particular, data is returned back to the requesting workstation identifying the drive type as a removable drive having, in the preferred embodiment, a total of 600 MBytes of available capacity.

Thus, it should be appreciated, that the emulated drives differ significantly from the actual physical drives in two respects. Firstly, the

20   emulated drives are significantly smaller than the actual physical drives on which they are being emulated, primarily to ensure that a large number of such drives may be supported by the system. Secondly, the physical drives are actually fixed drives and remain permanently in place. Thus, when the server writes data to a particular physical location, the server is assured that

25   this physical location will remain in place and will not be exchanged for some other data storage medium. However, in the emulated environment, the

requesting processors are informed that the drives to which they are writing should be treated removable drives, effectively warning the processor that these drives may be replaced and that a subsequent data transfer operation to that particular drive would not necessarily result in the same information being available on the storage medium.

In the system itself, the emulated drives are not physically replaced by other recording media and it is not actually necessary for a physical dismounting operation to be performed when data access has been completed. However, by informing the remote processors that they are dealing with removable disc drives, the resulting dismount or unload command issued by the operating systems of the remote processors will ensure that the server 20 has been instructed to the effect that the remote processors have completed their data transfer operations, thereby ensuring that the processor 20 receives sufficient information for it to complete its housekeeping tasks, thereby allowing other workstations to be given access to emulated drives once they have been released from a data transfer operation.

Thus, to summarise, when the server 20 grants access to an emulated logical disc drive, it informs the requesting processor that it has been given access to a removable disc drive having a total capacity of 600 MBytes.

Conventionally, data is written to disc drives as identifiable blocks. In order to optimise available storage space, these blocks would normally reside on physical drives as contiguous regions of storage, effectively reducing fragmentation. However, it is not essential for the data to be perceived as residing in contiguous regions. In the present embodiment, the workstation processors may write data to the logical disc drives as they feel fit. Thus a logical disc drive may be perceived as being fragmented.

Thus, at step 34 data transfer takes place and the workstation's local operating software may read and write to the logical drives as if they were local removable disc drives. However, given the nature of the RAID 5 drives 21 to 25, the rate of data transfer is substantially higher and only restricted by the capabilities of the interface circuits employed. Thus, as far as the workstation processor is concerned, along with its operating software, it is interfacing with a standard removable disc drive. However, as far as the actual operator is concerned, the rate of data transfer is significantly higher and, due to the parallel nature of the array, said transfer rate significantly exceeds that available from fast local hard drives. Thus, the operator is provided with the advantage of fast data access while at the same time allowing data to be shared between a plurality of users as if the data were contained on removable exchangeable drives. Furthermore, the physical removing and exchange of drives is not necessary and only occurs at a logical level.

After data transfer has been completed, a user will normally take measures to terminate access to the logical drive. Thus, at step 35, a user may request access to another drive or implement alternative local processing operations. In either event, the workstation operating system issues a dismount command to the server 20 at step 36. This dismount command is required when the operating system has been given access to real dismountable drives which, as previously stated, is acted upon by the server 20 so as to complete the housekeeping procedures.

At step 37 the server 20 acts upon the dismount command by releasing the logical drive such that it may be accessed by other workstations. Thereafter, at step 38, the server waits for the next user command.

The releasing of a logical drive will include updating the directory for that drive. In order to improve disc access speed, disc directories are cached in memory and directory updates are made locally while the processor has access to the disc. Upon receiving the dismount command, the updated
5   directory information from the cache memory will be rewritten back to the directory on the disc, thereby maintaining the integrity of the directory data stored on the disk.

The system operating the software will be aware of the way in which removable disc directories are handled and the system will include measures
10   for accommodating power failures and program errors etc. Thus, measures can be taken to effect a disc reset, upon detecting that a particular partition has become unavailable or disconnected, whereafter, when access has been regained in that particular drive, information will be read to the effect that no assumptions may be made about the data contained on the disc and it would
15   be necessary to re-assess that data.

Although the system emulates logical drives having, for example, 600 MBytes of available storage, physical space on the RAID 5 drives 21 to 25 is actually allocated dynamically in regions as storage space for the storage of actual data is required. Thus, although users appear to be given access to
20   logical drives having a total of 600 MBytes, space on the actual RAID 5 drives is not divided into 600 MByte partitions. Drives 21 to 25 are divided into blocks of between two and thirty-two MBytes and blocks are allocated dynamically as and when they are required.

The actual size of blocks on the RAID 5 drives may be variable,
25   although it will be assumed herein that, for a particular application, two MByte blocks will be identified. As data is written to a logical drive, via the

server 20, the data will physically occupy an identifiable two MByte block. As the volume of data increases beyond two MBytes, the server 20 will identify a new two MByte data block and data originating from the user will then be directed to this new block. Thus, if a user has created a total of five

5   MBytes, the server is required to maintain a list of where these five MBytes actually reside on the drives, in terms of three two MByte blocks. However, as far as the user is concerned, five MBytes of data have been written to on a removable drive having a total of 600 MBytes of available capacity.

At a workstation, a user is presented with the user interface capable of

10   providing an environment for allowing existing logical drives to be selected and for new logical drives to be defined. The user interface 61 is in turn supported by a local operating system 62, which is responsible for generating commands which are in turn interpreted by the interface.

As far as the local operating system 62 is concerned, access is being

15   made to a conventional SCSI disc drive and communication is effected over a conventional SCSI interface 63, resident at the workstation, to a server SCSI interface 65. This communication conforms to establish SCSI protocols, thereby substantially reducing the need for embedding bespoke software within the local workstation environments.

20   A server operating system 66 converts SCSI sector definitions into addressable physical data blocks by means of a look-up table, identified by reference 68. A look-up table is defined for each logical drive and when a logical drive is selected by an operator, its associated look-up table is loaded to an operating area of memory 28 within the server 20. Thus, within the

25   server operating system 66, a logical drive is identified, resulting in a table 68 being loaded. Thereafter, SCSI sector selections are supplied as inputs to

the table, which then results in addresses for physical data blocks being generated as outputs. Thus, as illustrated in Figure 4, the table 68 effectively points to addressable data blocks 69 in the array of physical data storing discs 21 to 25.

5   The server operating system 66 allows the SCSI environment of the user terminal to interface with the emulated environment of the server. Thus, it is necessary for the server operating system to emulate an SCSI disc drive and procedures for performing this emulation are detailed in Figure 5.

       The procedures shown in Figure 5 are executed within a multi-tasking

10   environment, such that similar procedures may be performed for each of the user terminals. The procedures shown in Figure 5 therefore represent instructions executed on behalf of a particular workstation.

       At step 71 the system waits for a workstation command and upon receiving such a command a question is asked at step 72 as to whether this

15   is a "mount" command. A "mount" command instructs the server to mount a selected removable drive and data transfers via the server 20 can only be performed if the server has received such an instruction. Thus, if the question asked at step 72 is answered in the negative, control is directed to step 73, whereupon procedures are performed to emulate an empty drive. Thus, this

20   would include the generation of error messages to the effect that the drive is not ready etc.

       If an instruction to mount a drive is generated by the workstation, the question asked at step 72 is answered in the affirmative, resulting in control being directed to step 74. At step 74 a question is asked as to whether the

25   drive is free and if another user workstation has been given access to that

particular drive, the question asked at step 74 will be answered in the negative, resulting in a reply being generated at step 75 to the effect that the drive is not ready. Thereafter, control is returned to step 71. However, if the drive is free the question asked at step 74 is answered in the affirmative, resulting in control being directed to step 76.

At step 76 a partition is identified representing the regions within which data for the emulated drive may be read from or written to. Thereafter, control is directed to step 77, whereupon a reply is returned back to the requesting workstation to the effect that the disk has been mounted and control is directed to step 78.

At step 78 the server waits for further commands from the user workstation and in response to receiving such a command, a question is asked at step 79 as to whether this is a dismount command. If the command is not a dismount command further emulation of a removable disc is performed at step 81 and control is returned to step 78.

Upon detecting a dismount command at step 79, control is directed to step 81, whereupon the partition is de-allocated and a reply is issued to the user workstation at step 82 to the effect that the disc has been dismounted. Thereafter control is returned to step 71, whereupon the server waits for the next workstation command.

CLAIMS

1. A method of storing data, wherein a large storage volume emulates a plurality of logical drives; said logical drives emulate removable disc drives; and the closing of access to a previously accessed logical drive generates a disc dismount command.

2. A method according to claim 1, wherein the logical drives emulate removable SCSI drives.

3. A method according to claim 2, wherein each of said logical drives provides between 200 MBytes and 900 MBytes of data storage.

4. A method according to any of claims 1 to 3, wherein data is written to the physical storage volume in identifiable blocks.

5. A method according to claim 4, wherein each of said blocks provides between one MByte and sixty-four MBytes of storage.

6. A method according to claim 4 or claim 5, wherein a mapping table maps sectors of an emulated disc onto blocks of the physical volume.

7. A method according to claim 4 or claim 5, wherein blocks are allocated dynamically as storage is required.

8. A method according to any of claims 1 to 7, wherein the storage volume is implemented as an array of disc storage devices.

9. A method according to claim 8, wherein the array has redundant discs.

10. A method according to claim 8 or claim 9, wherein the array has between four and twelve discs.

11. A method according to any of claims 1 to 10, wherein directory information stored on an accessed disc is updated in response to a disc dismount command.

12. A method according to any of claims 1 to 10, wherein directory information stored on an accessed disc is updated on detecting that a user terminal has been disconnected and can no longer access a previously accessed logical drive.

13. Data storage apparatus, including a large storage volume;

a control device arranged to control data transfer with said storage volume and to provide user terminal access to said storage volume by emulating the presence of a plurality of removable disc drives, wherein

user terminals generate a disc dismount command when closing access to a previously accessed logical drive; and

the control device responds to said disc dismount command by terminating connection to said previously connected logical drive.

14. Apparatus according to claim 13, wherein the logical drives emulate removable SCSI drives.

15. Apparatus according to claim 14, wherein each of said logical drives provides between 200 MBytes and 900 MBytes of data storage.

16.   Apparatus according to any of claims 13 to 15, wherein the control device is arranged to write data to the physical storage volume in the form of identifiable blocks.

17.   Apparatus according to claim 16, wherein each of blocks provides between 1 MByte and 64 Bytes of storage.

18.   Apparatus according to claim 16 or claim 17, wherein the control device is arranged to access mapping tables, mapping sectors of an emulated disc onto blocks of the physical volume.

19.   Apparatus according to any of claims 16 to 18, wherein the control device is arranged to dynamically allocate blocks as storage is required.

20.   Apparatus according to any of claims 13 to 19, where the storage volume is implemented as an array of disc storage devices.

21.   Apparatus according to claim 20, wherein the array includes redundant discs.

22.   Apparatus according to claim 20 or claim 21, wherein the array has between four and 12 discs.

23.   Apparatus according to any of claims 13 to 22, wherein the control device is arranged to read directory information from an accessed logical drive, and the directory information stored on the disc is updated in response to a disc dismount command.

24. Apparatus according to any of claims 13 to 22, wherein the control device is arranged to read directory information from an accessed logical drive and directory information stored on a logical disc drive is updated by the control device in response to detecting that a user terminal has been disconnected and can no longer access a previously accessed logical drive.

5

25. A method of storing data substantially as herein described with reference to the accompanying drawings.

26. A data storage apparatus substantially as herein described with reference to the accompanying drawings.

10

Application No: GB 9502377.6	**Examiner:** Geoff Western
Claims searched: 1-26	**Date of search:** 3 May 1995

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.N): G4A (AFS, AMX)

Int Cl (Ed.6): G06F (3/06)

Other: On-line : WPI, INSPEC, COMPUTER DATABASE

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
A	EP-0078683-A2 (FUJITSU) See whole document	-
A	Dialog record 01425541 of UNIX Review, vol 9, No 4, April 1991, page 98	-

X Document indicating lack of novelty or inventive step	A Document indicating technological background and/or state of the art.
Y Document indicating lack of inventive step if combined with one or more other documents of same category.	P Document published on or after the declared priority date but before the filing date of this invention.
	E Patent document published on or after, but with priority date earlier than, the filing date of this application.
& Member of the same patent family	

An Executive Agency of the Department of Trade and Industry